REMARKS

This current Reply is responsive to a current and Non-final Office Action dated (mailed) 12/15/2005. This current Office Action examined claims 1-26.

Generally, the current Office Action rejected claims 1-26.

Specifically, the current Office Action indicated the following:

Claims 1-22 and 24-26 are rejected under 35 U.S.C. 102(e) as being anticipated by Tanguay et al. (US 5,946,488).

Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over Tanguay et al. (US 5,946,488).

No claims are canceled or added by this Reply. Hence, claims 1-26 are presented for examination. Of claims 1-26, claims 1, 9, and 15 are independent.

Returned PTO-1449s

Applicants thank the Examiner for considering the cited documents and returning the submitted PTO-1449s with initials and signatures.

No initial or prima facie rejection has been established

at least for claims 5 and 9-14, including independent claim 9.

The entirety of the 35 U.S.C. 102(e) rejection from page 2 and page 3 of the current Office Action is reproduced below:

> As per claims 1-4, 6-8, 15-22, and 26, Tanguay teaches a system for processing command line input, the system comprising: a command line interface (user input, 230, display, 240, code viewer, 220) comprising a set of executable commands; and a command line processor for, at least; parsing the command line input; identifying one or more macros within the input, expanding the one or more macros into at least one executable command of the command line interface, and executing the commands independent of compilation. (Abstract, cols.2-12)
>
> Tanguay teaches the use computer programmer system wherein the user is able to selectively examine specific source code including the selective expansion of macro that appears in the source code. The user is able to expand the macro into other macro calls and also has the ability to expand and un-expand selected macro calls. The computer programmer system allows the expansion of the macro in a preprocessed system, in other words, the user is allowed to expand and un-expand selected macro calls before complication of the source code.

As is evident from a review of the above text, the current Office Action fails to address claims 5 and 9-14 in the rejection.

MPEP §2131 (Rev. 3, August 2005) reads on page 2100-76 in the right hand column, in pertinent part:

> TO ANTICIPATE A CLAIM, THE REFERENCE MUST TEACH EVERY ELEMENT OF THE CLAIM
>
> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

The Office does not allege that any of the specific elements, much less that each and every element, of claims 5 and 9-14 are taught by Tanguay et al. Consequently, the Office has failed to meet its initial burden. There has been no prima facie rejection established against the unique elements of any of claims 5 and 9-14.

Accordingly, withdrawal of the rejection of claims 5 and 9-14 is respectfully requested.

Tanguay et al. (U.S. Patent No. 5,946,488) is not related to, does not describe, and does not teach "command line interface".

A textual search of Tanguay et al. for "command line interface", "command line", "line interface", and even "command interface" was conducted. None of the terms could be found within Tanguay et al.

Moreover, the *concept* of the command line interface also does not appear to be present. The current Office Action, as more fully reproduced above, alleges that "Tanguay teaches a system for processing command line input, the system comprising: a command line interface (user input, 230, display, 240, code viewer, 220) ...". It is unclear to Applicants' representative how user input 230, display 240, and selective code viewer 220 (Tanguay et al.; Figure 2) can possibly individually and/or jointly correspond to a **command line interface** as claimed. The current Office Action does not appear to explain how such a correspondence is legally appropriate or technically accurate.

In contrast, Tanguay et al. appears to be directed to standard coding environments where a programmer can review, analyze, and debug code prior to compilation. See, e.g., Figure 3, which includes steps leading up to "normal compilation 324". Other examples of the environment of Tanguay et al. are:

> 1. Field of the Invention
>
> The present invention relates to compilers, assemblers and preprocessors for computer programs, and more specifically to an apparatus and a method for selectively and incrementally viewing the results of preprocessing operations such as macro expansions.

[Tanguay et al.; Column 1, Lines 7-12]

> Preprocessing translates original source code into actual source code that can be compiled or assembled.

[Tanguay et al.; Column 1, Lines 17-18]

> The present invention makes it possible for programmers to selectively and incrementally view the results of preprocessing operations such as macro expansion. The invention allows users to expand specific macros and other preprocessor directives, while continuing to view the rest of the source code in its original state. The present invention allows programmers to selectively switch back and forth from the original code to full or partial expansions of the code. This simplifies the process of debugging macros as well as the resulting compiled source code.

[Tanguay et al.; Column 1, Line 61 to Column 2, Line 3]

> One potential embodiment of the present invention is incorporated into software which serves as a code viewer and editor. Such software is often called an Integrated Development Environment (IDE).

[Tanguay et al.; Column 2, Lines 29-31]

> In one embodiment of the present invention, the preprocessed code that results from the preprocessing phase can be sent to any of three destinations: (1) the code can be sent to the selective viewing phase of the present invention; (2) the code can be sent to a "stripper" program which strips out the annotations. Once the stripper has done its work, the code can be sent to a normal compiler for compilation; and (3) the code can be sent directly to a compiler. In this case, the compiler has to strip out the annotations itself as it reads the processed code.

[Tanguay et al.; Column 8, Lines 1-10]

Thus, Tanguay et al. is not related to, does not describe, and does not teach "command line interface".

Each of independent claims 1, 9, and 15 recite an element of **command line interface**. Consequently, Tanguay et al. cannot anticipate claim 1, claim 9, or claim 15.

Accordingly, withdrawal of the rejection of independent claims 1, 9, and 15 is respectfully requested.

Tanguay et al. (U.S. Patent No. 5,946,488) neither describes nor teaches execution independent of compilation.

Claim 1 reads, in pertinent part, **executing the command independent of compilation**.

Claim 9 reads, in pertinent part, **executing the batch file, including the command, independent of compilation**.

Claim 15 reads, in pertinent part, **executing the commands independent of compilation**.

Tanguay et al. describes expanding preprocessor constructs in response to user input in the context of debuggers, compilation programs, and other development environments. However, it is respectfully submitted that Tanguay et al. does not teach any execution prior to compilation.

Furthermore, it does not appear that the current Office Action citers to any part of Tanguay et al. that addresses execution of commands independent of compilation. Thus, Tanguay et al. does not anticipate claim 1, claim 9, or claim 15.

Accordingly, withdrawal of the rejection of independent claims 1, 9, and 15 is respectfully requested.

Dependent Claims

Reasons for the allowability of independent claims 1, 9, and 15 have been provided above. Claims 2-8/22-26, 10-14, and 16-21 depend directly or indirectly from these independent claims 1, 9, and 15, respectively. Although each also includes additional element(s) militating toward allowability, these dependent claims are allowable at least for the reasons given above in connection with their respective independent claims.

Examples of such additional element(s) militating toward allowability include, by way of example only, those of claims 2, 3, 5, and 12 that involve prompting a user for input. Tanguay et al. only permits a user to expand or contract a macro. There is no teaching in Tanguay et al. regarding prompting a user for input. Hence, dependent claims 2, 3, 5, and 12 are separately allowable for at least this reason.

CONCLUSION

It is respectfully submitted that all of claims 1-26 are allowable. The Examiner is therefore respectfully requested to pass the instant Patent Application to issue.

Respectfully Submitted,

Date: 3/14/2005 By: Keith W. Saunders

Keith W. Saunders
Reg. No. 41,462
(509) 324-9256 x238